

IN THE CLAIMS:

1. (Currently Amended) A method of developing a computer software system, comprising the computer-implemented steps of:
 - defining a first interface ~~between~~ associated with a proposed view sub-system and with a proposed business logic sub-system, wherein the proposed view sub-system and the proposed business logic sub-system interact only via the first interface;
 - defining a second interface ~~between~~ associated with a proposed handler sub-system and with the proposed business logic sub-system, wherein the proposed handler sub-system and the proposed business logic sub-system interact only via the second interface;
 - wherein the proposed view sub-system, the proposed business logic sub-system, and the proposed handler sub-system are all isolated from each other;
 - creating the proposed view sub-system in accord with the first interface; and
 - creating the proposed handler sub-system in accord with the second interface.
2. (Currently Amended) The method according to claim 1, further comprising the steps of:
 - defining a third interface ~~between~~ associated with the proposed view sub-system and with the proposed handler sub-system; and
 - creating the proposed view sub-system in accord with both the first and third interfaces.
3. (Currently Amended) The method according to claim 1, further comprising the steps of:
 - defining a fourth interface ~~between~~ associated with the proposed view sub-system and with the proposed handler sub-system; and
 - creating the proposed handler sub-system in accord with both the second and the fourth interfaces.
4. (Currently Amended) The method according to claim 1, further comprising the steps of:

defining a third interface ~~between~~ associated with the proposed view sub-system and the proposed handler sub-system;
defining a fourth interface ~~between~~ associated with the proposed view sub-system and with the proposed handler sub-system;
creating the proposed view sub-system in accord with both the first and third interfaces; and
creating the handler sub-system in accord with both the second and the fourth interfaces.

5. (Original) The method according to claim 1, wherein:
the first interface defines a plurality of methods for data storage and retrieval that are implemented in the business logic sub-system.
6. (Original) The method according to claim 1, wherein:
the second interface defines a plurality of methods of business logic that are implemented in the business logic sub-system.
7. (Original) The method according to claim 2, wherein:
the third interface is a listener interface that defines a plurality of methods in the handler sub-system which respond to actions in the view sub-system.
8. (Original) The method according to claim 3, wherein:
the fourth interface defines a plurality of methods which are implemented in the view sub-system for use by the handler sub-system.
9. (Original) The method according to claim 1, wherein:
the view sub-system includes a plurality of user interface objects;
the handler sub-system includes a plurality of use case control objects; and
the business logic sub-system includes a plurality of business logic objects.
10. (Original) The method according to claim 1, wherein:
the sub-systems are created substantially independently of each other once the interfaces have been defined.

11. (Currently Amended) A computer software system in a computer readable medium, said system comprising:

first instructions defining a view sub-system including presentation objects which provide a user interface;

second instructions defining a business logic sub-system including use case objects which hold business data and implement business functions;

third instruction defining a handler sub-system including controller objects which control actions of the view sub-system and actions of the business logic sub-system;

fourth instructions defining a data interface only through which the view sub-system obtains business data for the presentation objects; and

fifth instructions defining a business interface only through which the handler sub-system invokes business functions.

12. (Currently Amended) The system according to claim 11, further comprising:

sixth instructions defining a listener interface through which the handler sub-system responds to events in the user interface.

[[12]] 13. (Currently Amended) The system according to claim 11, further comprising:

sixth instructions defining a view action interface through which the handler sub-system invokes actions in the user interface.

[[13]] 14. (Currently Amended) A computer program in a computer readable medium, said program comprising:

first instructions defining at least one view object including presentation objects which provide a user interface;

second instructions defining at least one business logic object holding business data and implementing business functions;

third instructions defining at least one handler object which controls actions of at least one of the view objects and actions of at least one of the business logic objects;

fourth instructions defining a data interface only through which the at least one view object obtains business data for the presentation objects; and
fifth instructions defining a business interface only through which the at least one handler object invokes business functions.

[[14]] 15. (Currently Amended) The computer program according to claim [[13]] 14, further comprising:

sixth instructions defining a listener interface through which the handler object responds to events in the user interface.

[[15]] 16. (Currently Amended) The computer program according to claim [[13]] 14, further comprising:

sixth instructions defining a view action interface through which the handler object invokes actions in the user interface.

17. (New) The computer program system of claim 11 further comprising:
sixth instructions for further defining the view sub-system, the business logic sub-system, and the handler sub-system such that each sub-system is isolated from another sub-system.

18. (New) The computer program of claim 14 further comprising:
sixth instructions for further defining the view sub-system, the business logic sub-system, and the handler sub-system such that each sub-system is isolated from another sub-system.